UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/760,405	01/12/2001	Xiaoning Nie	GR 00 P 1031	1766

24131 7590 02/22/2007

LERNER GREENBERG STEMER LLP
P O BOX 2480
HOLLYWOOD, FL 33022-2480

EXAMINER	
LI, AIMEE J	
ART UNIT	**PAPER NUMBER**
2183	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	02/22/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No. 09/760,405	Applicant(s) NIE, XIAONING	
	Examiner Aimee J. Li	Art Unit 2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on 18 September 2006.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1 and 4-15 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1 and 4-15 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on 12 January 2001 is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1, 4-12, and new claims 13-15 have been considered. Claim 1 has been amended as per Applicant's request. New claims 13-15 have been added as per Applicant's request.

2. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 18 September 2006 has been entered.

Papers Submitted

3. It is hereby acknowledged that the following papers have been received and placed of record in the file: RCE as filed 18 September 2006; Amendment as filed 18 September 2006; and Extension of Time for 1 Month as filed 18 September 2006.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1, 4-11, and 13-15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Suzuki, U.S. Patent Number 6,499,096 (herein referred to as Suzuki) in view of Hirata et al., "An Elementary Processor Architecture with Simultaneous Instruction Issuing from Multiple Threads" ©1992 ACM (herein referred to as Hirata).

6. Referring to claims 1 and 13-14, taking claim 1 as exemplary, Suzuki has taught a data-processing device for processing in parallel a plurality of processes, comprising:

a. A multiplicity of bundles with a plurality of instructions of a process, the instructions of a bundle being executable in parallel (Suzuki column 2, lines 14-29 "For this operation, FM (Format-Specifying) bits are assigned to lower order two bits…");

b. A branching control unit connected to and addressing said program memory a register for storing flags and data which are switched in dependence on a process being executed (Suzuki column 6, lines 29-44 ""…a branch target register (BTR) **10**…" and 52-67 "…In writing operation to program count **9** shown in FIG. **3**, it is controlled so that the output of branch target register **10** is given priority over the output of incrementer **13**…"; column 12, line 36 to column 13, line 5 "…a branch target register (BTR) **76**…"; Figure 3; and Figure 11);

c. A program flow control unit connected to said branching control unit (Suzuki column 6, lines 29-44 ""…a branch target register (BTR) **10**…" and 52-67 "…In writing operation to program count **9** shown in FIG. **3**, it is controlled so that the output of branch target register **10** is given priority over the output of incrementer **13**…"; and Figure 3), said program flow control unit controlling a fetching of bundles to be processed in parallel from said program memory, controlling said branching control unit, and controlling an output of instructions to be processed in parallel in dependence on information contained in the instructions and included in a compiling time of the program (Suzuki column 6, lines 29-44 ""…a branch

target register (BTR) **10**..." and 52-67 "...In writing operation to program count **9** shown in FIG. **3**, it is controlled so that the output of branch target register **10** is given priority over the output of incrementer **13**..."; and Figure 3);

d. A number N of instruction buffers being connected in parallel downstream of said program memory for storing instructions read out from said program memory (Suzuki column 4, line 56 to column 5, line 6 "A 'left container' refers to the left one of instruction buffers...A 'right container' refers to the right one of instruction buffers..."; column 12, lines 7-13 "...this embodiment has the structure formed by connecting a plurality of the VLIW processors in the first embodiments...the VLIW type processor in this embodiment is a 2n-way VLIW processor..."; column 12, line 36 to column 13, line 5 "...selectors **87** to **90** select the outputs of instruction memory **1**..."; Figure 3; Figure 9; and Figure 11), an instruction bundle being read into one of said instruction buffers and a second instruction bundle associated with a different process being read into another one of said instruction buffers (Suzuki column 4, line 56 to column 5, line 6 "A 'left container' refers to the left one of instruction buffers...A 'right container' refers to the right one of instruction buffers..."; column 12, lines 7-13 "...this embodiment has the structure formed by connecting a plurality of the VLIW processors in the first embodiments...the VLIW type processor in this embodiment is a 2n-way VLIW processor..."; column 12, line 36 to column 13, line 5 "...selectors **87** to **90** select the outputs of instruction memory **1**..."; Figure 3; Figure 9; and Figure 11); and

e. An instruction output selector being connected to and controlled by said process flow control unit causing said instruction output selector to read out instructions from said instruction buffers and output N instructions in parallel (Suzuki column 7, line 62 to column 8, line 24 "Exchanging portion **8'** includes selectors **21** and **22**…"; column 12, line 36 to column 13, line 6 "…an exchanging portion **74** which exchanges the contents of first to fourth containers…"; Figure 3; and Figure 11).

7. Suzuki has not explicitly taught

a. A program memory having stored therein at least one compiled program with a multiplicity N of independent processes, the compiled program including information on parallelism; and

b. Said instruction output selector having a multiplexer logic and selecting based on a determination by said program flow control unit, in a first case, one instruction from a first instruction buffer and one instruction from a second instruction buffer for execution in parallel and, in a second case, two instructions from one of said first and second instruction buffers for execution in parallel, said program flow control unit choosing between said first case or said second base based on at least one of priority information, NOP information and information of the current process.

8. However, Suzuki has taught a processor for improved performance in image processing (Suzuki column 1, lines 18-20 "There have been increasing demands for improved processor performance in various fields such as…high resolution image processing.") selecting output from

multiple buffers for processing (Suzuki column 7, line 62 to column 8, line 24 "...In the sub cycle, control system **14** controls selector **21** to select a sub-instruction, which belongs to the instruction sequence for right sub-instructions...and selector **22** to select a sub-instruction..."; column 12, line 36 to column 13, line 6 "...selectors **87** to **90** select the outputs of instruction memory..."; Figure 3; and Figure 11). Hirata has taught a processor for graphics (Hirata section 1 Introduction, paragraphs 1-2 "...However, the generation of high quality images requires great processing power...we present a processor architecture[1] used as the base processor in a parallel machine which could run such a graphics system) with

a. A program memory having stored therein at least one compiled program with a multiplicity N of independent processes, the compiled program including information on parallelism (Hirata section 1 Introduction, paragraph 2 "...In these algorithms, intersection tests account for a large part of processing time for the whole program. This test has inherent coarse-grained parallelism..."; section 2.3.1 (Parallel Execution of a Single Loop) Overview "Parallel execution of a single loop is available on our multithreaded machine..." and section 2.3.2 Static Code Scheduling "...Our code scheduler, however, checks entries..."); and

b. Said instruction output selector having a multiplexer logic and selecting based on a determination by said program flow control unit, in a first case, one instruction from a first instruction buffer and one instruction from a second instruction buffer for execution in parallel and, in a second case, two instructions from one of said first and second instruction buffers for execution in parallel (Hirata section 1 Introduction, paragraphs 3-5 "...parallel multi-threading within a

processor...When an instruction from a thread is not able to be issued...an independent instruction from another thread is executed..."; section 2 Processor Architecture "As shown in Figure 2, the processor is provided with several instruction queue unit and decode unit pairs, called *thread slots*. Each thread slot, associated with a program counter, makes up a *logical processor*..."; Figure 1; and Figure 2), said program flow control unit choosing between said first case or said second base based on at least one of priority information (Hirata section 2.2 Instruction Scheduling Strategy "...A unique priority is assigned to each thread slot..."), NOP information (Hirata section 2.3.2 Static Code Scheduling "When all of the instructions without dependencies at an issuing cycle have resource conflicts, a software pipeline would generate a NOP code...") and information of the current process (Hirata section 1 Introduction, paragraph 3 "When an instruction from a thread is not able to be issued because of either a control or data dependency within the thread, an independent instruction from another thread is executed." and section 2.3.2 "When all of the instructions without dependencies at an issuing cycle have resource conflicts, a software pipeline would generate a NOP code...").

9. In regards to Hirata, the general conventional system in Hirata's Figure 1(a) is similar to that shown in detail by Suzuki. As Hirata teaches in section 1 Introduction, paragraph 3, conventionally organized systems only utilizes the busiest functional unit about 30% of the time due to instruction dependencies found within an individual thread. By allowing another thread's instructions to execute on a functional unit when it is not being utilized, utilization is increased to

90%. Hence, Hirata's parallel multithreading fetches instructions either from a single thread or from different threads (Hirata section 1 Introduction, paragraph 3 "On the other had, parallel multithreading within a processor is a latency-hiding technique…When an instruction from a thread is not able to be issued because of either a control or data dependency within the thread, an independent instruction from another thread is executed.") stored in separate instruction queue units (Hirata section 2.1.1 Hardware Organization, paragraph 1 "As shown in Figure 2, the processor is provided with several instruction queue unit and decode unit pairs, called *thread slots*. Each thread slot, associated with a program counter, makes up a *logical processor…*"), e.g. first and second buffers, based upon a scheduling strategy that takes into account thread priorities (Hirata section 2.2 "Figure 4 illustrates an example of the policy with multi-level priorities." and Figure 4), NOP information (Hirata section 2.3.2 "When all of the instructions without dependencies at an issuing cycle have resource conflicts, a software pipeline would generate a NOP code…"), and dependencies and resource conflicts (Hirata section 1 Introduction, paragraph 3 "When an instruction from a thread is not able to be issued because of either a control or data dependency within the thread, an independent instruction from another thread is executed." and section 2.3.2 "When all of the instructions without dependencies at an issuing cycle have resource conflicts, a software pipeline would generate a NOP code…"), e.g. information of the current process. A person of ordinary skill in the art at the time the invention was made would have recognized, and as taught by Hirata, that incorporating simultaneous instruction issuing from multiple threads "achieves an efficient and cost-effective processor design which is oriented specifically for use as an elementary processor in a large scale multiprocessor system (Hirata section 1 Introduction, paragraph 4)" and optimizes utilization of

functional units (Hirata section 1 Introduction , paragraph 4 "The idea of parallel multithreading arose from hardware resource optimization..."). Therefore, it would have been obvious to a person of ordinary skill in the art the time the invention was made to incorporate the simultaneous instruction issuing from multiple threads of Hirata in the device of Suzuki to increase a processors efficiency and cost-effectiveness via functional unit utilization optimization.

10. Claims 13-14 contain similar limitations to claim 1 and are rejected for similar reasons. Claim 13 differs in that selection is based upon NOP instruction conditions. Claim 14 differs in that selection is based upon priority conditions. Both of which are claimed in the alternative in claim 1, however, the Examiner has shown how each of these limitations are rejected under Suzuki in view Hirata.

11. Referring to claim 4, Suzuki in view of Hirata has taught which comprises N instruction decoders for decoding the instructions being output (Suzuki column 4, line 56 to column 5, line 6 "...A 'left instruction decoder'...A 'right instruction decoder'...); column 12, lines 7-13 "...the structure formed by connecting a plurality of the VLIW processors..."; column 12, line 36 to column 13, line 5 "...a first instruction decoder **66**, a second instruction decoder **67**..."; Figure 3; Figure 9; and Figure 11).

12. Referring to claim 5, Suzuki in view of Hirata has taught at least two instruction-execution units for outputting the N decoded instructions (Suzuki column 4, line 56 to column 5, line 6 "...an instruction which is predetermined to be processed in the left processing unit...an instruction which is predetermined to be processed in the right processing unit..."; column 12, lines 7-13 "...the structure formed by connecting a plurality of the VLIW processors..."; column

12, line 36 to column 13, line 5 "...a first processing unit **70**, a second processing unit **71**...";
Figure 3; Figure 9; and Figure 11).

13. Referring to claim 6, Suzuki in view of Hirata has taught a data memory and at least two buses connecting said N instruction-execution units to said data memory (Hirata section 2 Processor Architecture "...A large general-purpose register file...Each bank has two read ports and one write port..." and Figure 2).

14. Referring to claim 7, Suzuki in view of Hirata has taught a plurality of instruction-execution units connected to said program flow control unit and configured to execute the instructions of one or more bundles in parallel (Suzuki column 4, line 56 to column 5, line 6 "...an instruction which is predetermined to be processed in the left processing unit...an instruction which is predetermined to be processed in the right processing unit..."; column 12, lines 7-13 "...the structure formed by connecting a plurality of the VLIW processors..."; column 12, line 36 to column 13, line 5 "...a first processing unit **70**, a second processing unit **71**..."; Figure 3; Figure 9; and Figure 11).

15. Referring to claim 8, Suzuki in view of Hirata has taught wherein said branching control unit is configured to output an address pointer for addressing a bundle (Suzuki column 6, lines 29-44 """...a branch target register (BTR) **10**..." and 52-67 "...In writing operation to program count **9** shown in FIG. **3**, it is controlled so that the output of branch target register **10** is given priority over the output of incrementer **13**..."; and Figure 3)

16. Referring to claim 9 , Suzuki in view of Hirata has taught wherein the branching control unit comprises:

a. A first multiplexer and a second multiplexer (Suzuki column 6, lines 29-44 "...selectors **11** and **12**..." and 52-67 "...control system **14** controls so that selector **11** selects the outputs of program counter..."; column 12, line 36 to column 13, line 5 "...a branch target register (BTR) **76**, selectors **77** and **78**..."; Figure 3; and Figure 11);

b. An adder (Suzuki column 6, lines 29-44 "...an incrementer **13**..." and 52-67 "...the output of the branch target register **10** is given priority over the output of incrementer **13**..."; column 12, line 36 to column 13, line 5 "...an incrementer **79**..."; Figure 3; and Figure 11);

c. N program counters (Hirata section 2 Processor Architecture "...Each thread slot, associated with a program counter..." and Figure 2); and

d. Wherein said program flow control unit feeds a number of instructions in a bundle to said adder and said adder adds an address pointer and the number of instructions (Suzuki column 6, lines 29-44 "...an incrementer **13**..." and 52-67 "...the output of the branch target register **10** is given priority over the output of incrementer **13**..."; column 12, line 36 to column 13, line 5 "...an incrementer **79**..."; Figure 3; and Figure 11);

e. Wherein said program flow control unit feeds addresses for program jumps or function calls and a process number to said first multiplexer (Suzuki column 6, lines 29-44 "...selectors **11** and **12**..." and 52-67 "...control system **14** controls so that selector **11** selects the outputs of program counter..."; column 12, line 36 to

column 13, line 5 "...a branch target register (BTR) **76**, selectors **77** and **78**..."; Figure 3; and Figure 11);

f. Said first multiplexer writing either the output signal of said adder or the addresses for program jumps or function calls into said program counter assigned to the active process (Suzuki column 6, lines 29-44 "...selectors **11** and **12**..." and 52-67 "...control system **14** controls so that selector **11** selects the outputs of program counter..."; column 12, line 36 to column 13, line 5 "...a branch target register (BTR) **76**, selectors **77** and **78**..."; Figure 3; and Figure 11); and

g. A content of said program counter assigned to the currently active process is output as a new address pointer via said, second multiplexer which is controlled using the process number supplied (Suzuki column 6, lines 29-44 "...selectors **11** and **12**..." and 52-67 "...control system **14** controls so that selector **11** selects the outputs of program counter..."; column 12, line 36 to column 13, line 5 "...a branch target register (BTR) **76**, selectors **77** and **78**..."; Figure 3; and Figure 11).

17. Referring to claim 10, Suzuki in view of Hirata has taught wherein said program flow control unit is configured to receive via a subbus of an output bus of said program memory at least one of the following:

a. At least one bit for indicating the parallel execution of instructions (Suzuki column 2, lines 14-29 "For this operation, FM (Format-Specifying) bits are assigned..."). In regards to Suzuki, the at least one bit is inherent for the instruction to be recognized as a VLIW instruction.

b. At least one bit for indicating the length of the following instruction bundle (Suzuki column 4, line 56 to column 5, line 6 "...A 'left sub-instruction'...A 'right sub-instruction'..."; column 12, lines 7-13 "...the structure formed..."; column 12, line 36 to column 13, line 5 "...a program counter..."; Figure 3; Figure 9; and Figure 11). In regards to Suzuki, the at least one bit is inherent for the instruction to recognize where it ends, since the instructions contains multiple sub-instructions.

c. The indication of one or more NOPs in the instruction bundles (Suzuki column 3, lines 49-52 "Since the compressing portion compresses the consecutive VLIW instructions based on a nop instruction...");

d. A priority of the processes of the instructions (Hirata section 2.2 "Figure 4 illustrates an example of the policy with multi-level priorities." and Figure 4).

18. Referring to claim 11, Suzuki in view of Hirata has taught wherein a process is called with a run instruction assigning a process number (Hirata section 2.1.3 Support of Concurrent Multithreading "...An instruction address save register and a thread status register are used as save areas for the program counter and program status words..."), a priority (Hirata section 2.2 "Figure 4 illustrates an example of the policy with multi-level priorities." and Figure 4) and a memory address of a starting point of the process in the program memory (Hirata section 2.1.3 Support of Concurrent Multithreading "...An instruction address save register and a thread status register are used as save areas for the program counter and program status words...").

19. Referring to claim 15, Suzuki in view of Hirata has taught at least two instruction-execution units for receiving the instructions selected by said instruction output selector, each of

said at least two instruction-execution units being able to execute instructions output by either of said first instruction buffer and said second instruction buffer (Hirata section 2 Processor Architecture "...Unless an instruction conflicts with other instructions issued from other decode units over the same functional unit, the instruction is sent immediately to the functional unit and executed..." and Figure 2).

20. Claim 12 is rejected under 35 U.S.C. 103(a) as being unpatentable over Suzuki, U.S. Patent Number 6,499,096 (herein referred to as Suzuki) in view of Hirata et al., "An Elementary Processor Architecture with Simultaneous Instruction Issuing from Multiple Threads" ©1992 ACM (herein referred to as Hirata), as applied to claim 1 above, and further in view of Allen, Jr. et al., U.S. Patent No. 6,404,752 (herein referred to as Allen). Suzuki in view of Hirata has not explicitly taught wherein said data processing device is a network processor for processing layer 1 to 7 of protocol stacks in applications including LAN, ATM switches, IP routers, and frame relays based on a system selected from the group consisting of DSL, Ethernet, and cable modems. Allen has taught the use of general-purpose microprocessors as network processors to provide a cost-effective solution to processing protocol stack layers for IDSN, cable and DSL modems that provides high throughput and speeds (Allen column 1, lines 44-49 "A typical architecture based upon the ISO model extends from Layer 1..." and column 2, line 38 to column 3, line 23 "...Industry consultants have defined a network processor..."). One of ordinary skill in the art would have recognized that a primary goal in microprocessor design is to lower costs while maintaining a high level of performance. Therefore, one of ordinary skill in the art would have found it obvious to modify the processor of Suzuki in view of Hirata to be

used in a network processor to be keep costs low while providing a high level of network processing performance.

Response to Arguments

21. Applicant's arguments with respect to claims 1 and 4-15 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

22. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

a. Holt, U.S. Patent Number 5,530,816, has taught a parallel multithreaded system with individual thread buffers and decoders and shared execution units and memory.

b. Kimura et al., U.S. Patent Number 5,546,593, has taught a parallel multithreaded system with individual thread buffers and decoders and shared execution units and memory with scheduling based on a priority and resource dependency scheme.

c. Keckler et al., U.S. Patent Number 5,574,939, has taught a multi-threaded VLIW system with individual thread buffers.

d. Ito et al., U.S. Patent Number 5,742,782, has taught a multithreaded VLIW system with individual decoders and shared execution units.

e. Dubey et al., U.S. Patent Number 5,812,811, has taught a parallel multithreaded system with individual program counters, instruction buffers and shared functional units and memory.

f. Dutton et al., U.S. Patent Number 5,944,816, has taught a multiple context system with shared execution units.

g. Kimura et al., U.S. Patent Number 6,105,127, has taught a multithreaded processor with multiple decode units and functional units.

h. Dowling, U.S. Patent Numbers 6,170,051 and 6,363,475, have taught parallel multithreaded VLIW system.

i. Tullsen, Eggers, and Levy's "Simultaneous Multithreading: Maximizing On-Chip Parallelism" ©1995 ACM and Tullsen, Eggers, Emer, Levy, Lo, and Stamm's "Exploiting choice: instruction fetch and issue on an implementable simultaneous multiprocessor" ©1996 ACM, "Simultaneous Multithreading: A Platform for Next Generation Processors" ©1997 IEEE, and "Converting Thread-Level Parallelism to Instruction-Level Parallelism via Simultaneous Multithreading" ©1997 ACM have all taught simultaneous multithreading which issues multiple instruction from one or more threads every cycle with scheduling scheme.

j. Hily and Seznec's "Standard Memory Hierarchy Does Not Fit Simultaneous Multithreading" ©1998 has taught a simultaneous multithreading system with shared memory.

23. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Aimee J. Li whose telephone number is (571) 272-4169. The examiner can normally be reached on M-T 7:00am-4:30pm.

24. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

25. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Aimee J. Li

Aimee J. Li
15 February 2007